UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/621,227	07/15/2003	Joel K. Young	977.056US1	1216

21186 7590 10/23/2009

SCHWEGMAN, LUNDBERG & WOESSNER, P.A.
P.O. BOX 2938
MINNEAPOLIS, MN 55402

EXAMINER	
INGVOLDSTAD, BENNETT	

ART UNIT	PAPER NUMBER
2427	

NOTIFICATION DATE	DELIVERY MODE
10/23/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

uspto@slwip.com
request@slwip.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/621,227
Filing Date: July 15, 2003
Appellant(s): YOUNG, JOEL K.

Mr. Paul Urbanski
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 9 July 2009 appealing from the Office action mailed 12 February 2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The following are the related appeals, interferences, and judicial proceedings known to the examiner which may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal:

Application 10/621,153: Notice of Appeal filed 4 June 2009, Appeal Brief filed 4 August 2009.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,763,377	Belknap et al.	7-2004
6,763,501	Zhu et al.	7-2004

Synchronized Multimedia Integration Language (SMIL 2.0): W3C Recommendation 07 August 2001. http://www.w3.org/TR/2001/REC-smil20-20010807/.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1, 2, 5, 6, 8-19, 23-26, and 28-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Belknap (Patent 6,763,377) in view of Zhu (US Patent 6,763,501) and the SMIL playlist format (SMIL 2.0 W3C Recommendation).

For **Claim 1** Belknap teaches:

a system (Figure 1 with Column 8 Lines 12-15), comprising:

at least one video display (Col. 11 Lines 41-45; also Fig. 2 Element 62, and Col. 10 Lines 54-58);

at least one media server (Fig. 1 Elem. 14), each media server to communicate with one or more of the at least one video display (Col. 10 Lines 30-34; also Fig. 2 Elem. 62, and Col. 10 Lines 54-58);

at least one video file server (Fig. 1 Elem. 18), each video file server including a number of video files (Col. 8 Lines 31-34), each video file including video content to be selectively displayed on the at least one video display (Col. 4 Line 67, and Col. 5 Lines 1-5);

a web client (Fig. 1 Elem. 12) to communicate with each video file server through a network (Fig. 1) to configure at least one playlist in the video file server (Fig. 32 Elem. 1200, also Col. 44 Lines 46-58), each playlist including at least one track, wherein the track includes an identifier to select one or more of the number of video files… (Col. 5, Lines 6-9, also Fig. 32 Elements 1208 and 1214);

each video file server being configured to push video content (Fig. 32 Element 1240) from a selected video file in the video file server (Fig. 32 Elements 1208 and 1214) to a selected media server based on the playlist (Fig. 32 Elem. 1240); and

each media server to translate the pushed video content into a video output signal suitable for display on the video display (Col. 1 Lines 30-34, and Col. 8 Lines 66-67 through Col. 9 Lines 1-2).

Belknap does not expressly teach:

wherein each video file server includes a virtual display driver, that appears to be a video display to the video file server, to translate video content into application independent video content, thereby not requiring the media server to decode pushed video content;

Zhu teaches:

an application and document server (Fig. 2 Elements 106 and 216 with Col. 2 Lines 4-17) includes a virtual display driver (Fig. 4 Elem. 406 with Col. 4 Lines 58-67 through Col. 5 Lines 1-3), that appears to be a video display to the server (Col. 5 Lines 65-67 through Col. 6 Lines 1-6), to translate video content into application independent video content (Col. 4 Lines 65-67, and Col. 5 Lines 1-7, note Zhu teaches using a virtual device to load a document to be sent to a client for viewing), thereby not requiring the media server (Fig. 1 Elem. 102) to decode pushed video content (Col. 6 Lines 20-25, note Zhu teaches that Elem. 102 passes documents directly between the server and client devices, with no teaching of decoding documents within Elem. 102)

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate a virtual display driver as taught by Zhu, within the video file server taught by Belknap, in order to provide ubiquitous access to media for multiple client devices (Zhu: Col. 1 Lines 59-64), by providing application independent video content that does not need to be decoded by a media server.

Belknap in view of Zhu does not expressly teach that the track includes at least one logical action related to playing the playlist.

The SMIL specification describes a playlist format (see the playlist in section 11.1.1) wherein playlist tracks may include logical actions related to playing the playlist (11.1.1 playlist: a duration attribute comprises play and stop actions. See also accelerate, decelerate, autoReverse, and speed attributes in section 11.1.2).

The simple substitution of one known element for another to achiever predictable results is obvious. Therefore the simple substitution of the SMIL playlist format replacing the playlist format disclosed in Belknap would have been obvious, because such a substitution would have produced predictable results in view of the SMIL format's utility for playing videos according to a playlist, as required by the video server of Belknap in view of Zhu.

For **Claim 2**, beyond what was discussed in independent Claim 1, Belknap further teaches:

the system of Claim 1, wherein each media server further serves as a conversion agent to translate optionally pushed (Fig. 22 with Col. 32 Lines 23-26 and 48-56 describes optionally pushed media) application specific video content into a video output signal suitable for display (Col. 10 Lines 39-43, note Element 14 "natively understands the encoding format of the asset" implying appropriate conversion of application specific video to an appropriate display format).

For **Claim 5**, beyond what was discussed in Claim 4, Belknap further teaches:

the logical actions execute in the video file server as a decision tree (Col. 44 Lines 46-57, note the multicasting commands downloaded to the media server [such as Element 864 in Fig. 32] are fundamentally encoded in software as a series of logical decisions, which can be interpreted as a decision tree).

For **Claim 6**, beyond what was discussed in Claim 5, Belknap further teaches:

the video server executes the at least one playlist based on the logical actions, and wherein the logical actions are configured at least in part by the web client (Col. 44 Lines 33-57).

For **Claim 8**, beyond what discussed in Claim 6, Belknap in view of Zhu teaches:

a video server which executes a playlist based on logical actions (Belknap: Col. 44 Lines 33-57)

The preceding combination of Belknap in view of Zhu does not teach:

the system of Claim 6, wherein logical actions further include inputs external to the video file server

However, Zhu also teaches:

a remote client can access and manipulate a media object being presented by an application on a remote server (Col. 2 Lines 26-51)

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include inputs external to the media server as

taught by Zhu, as logical actions controlling the execution of the playlist within the video file server, in order to manipulate the execution of the playlist from an external location.

For **Claim 9**, beyond what was discussed in Claim 1, Belknap further teaches:

the logical actions further include a timed duration of playing the files (Fig. 32 Element 1242; also Fig. 22 Element 814 and Col. 32 Lines 55-60; with Col. 41 Lines 20-23).

For **Claim 10**, beyond what was discussed in Claim 1, Belknap further teaches:

the logical actions further include a time to initiate playing the files (Fig. 32 Elements 1230 and 1236 and 1238, and Col. 42 Lines 35-38).

For **Claim 11**, beyond what was discussed in Claim 1, Belknap further teaches:

the logical actions further include a time to terminate playing the files (Fig. 22 Elements 814 and 866; also Fig. 24 with Col. 36 Lines 38-42; and Col. 41 Lines 20-23).

For **Claim 12**, beyond what was discussed in Claim 1, Belknap further teaches:

the logical actions further include a number of times to play the files (Fig. 32 Elements 1242, 860, 862, and Col. 33 Lines 11-15).

For **Claim 13**, beyond what was discussed in Claim 8, Belknap in view of Zhu teaches:

the system of Claim 8, wherein the inputs external to the video file server are mapped into application specific commands according to the format of the video file (Zhu: Fig. 2 Element 212 with Col. 4 Lines 33-39, and Col. 5 Lines 1-13).

For **Claim 14**, beyond what was discussed in independent Claim 1, Belknap further teaches:

the video file further includes audio content (Col.1 Lines 17-20, and Col. 8 Lines 30-35).

For **Claim 15**, beyond what was discussed in independent Claim 1, Belknap further teaches:

the video content includes any combination from the set of Power Point, J-Peg, Video Clip, or Web formats (Col. 8 Lines 30-35, note that web format includes html which is text based; also Col. 10 Lines 30-35).

For independent **Claim 16** Belknap teaches:

a video file server (Fig. 1 Elem. 18), comprising:

memory to store video files (Fig. 1 Elem. 22, also Col. 8 Lines 30-35) and at least one playlist (Col. 44 Lines 45-54), each video file including video content to be selectively displayed on at least one video display (Col. 4 Line 67, and Col. 5 Lines 1-5), each playlist including a list of identifiers for video files (Col. 5 Lines 6-9, also Fig. 32 Elements 1208 and 1214), a file server location of the video files (Col. 5 Lines 6-9; also Fig. 32 Elements 1202, 1204, 1208 and 1214; and Col. 44 Lines 45-54), and logical actions related to playing the selected video content (Fig. 32 Element 1200, and Col. 44 Lines 45-54); and

a processor executing application specific software to push the selected video content according to the playlist to at least one media server (Fig. 2 Elem. 14) for display (Col. 8 Lines 54-65; and Fig. 32 Element 1200, and Col. 44 Lines 45-54).

Belknap does not expressly teach:

wherein the processor includes a virtual display driver, configured to translate video content into application independent video content, thereby not requiring the media server to decode pushed video content;

Zhu teaches:

a software program running on a processor (Fig. 2 Elements 106 and 216, with Col. 4 Lines 53-57) includes a virtual display driver (Fig. 4 Elem. 406 with Col. 4 Lines 58-67 through Col. 5 Lines 1-3), configured (Col. 5 Lines 65-67 through Col. 6 Lines 1-6), to translate video content into application independent video content (Col. 4 Lines 65-67, and Col. 4-7, note Zhu teaches using a virtual device to load a document to be sent to a client for viewing), thereby not requiring the media server (Fig. 1 Elem. 102) to decode pushed video content (Col. 6 Lines 20-25, note Zhu teaches that Elem. 102 passes documents directly between the server and client devices, with no teaching of decoding documents within Elem. 102).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the virtual display driver taught by Zhu, within the video file server taught by Belknap, in order to provide ubiquitous access to media for multiple client devices (Zhu: Col. 1 Lines 59-64), by providing

application independent video content that does not need to be decoded by a media server.

Belknap in view of Zhu does not expressly teach that the track includes at least one logical action related to playing the playlist.

The SMIL specification describes a playlist format (see the playlist in section 11.1.1) wherein playlist tracks may include logical actions related to playing the playlist (11.1.1 playlist: a duration attribute comprises play and stop actions. See also accelerate, decelerate, autoReverse, and speed attributes in section 11.1.2).

The simple substitution of one known element for another to achiever predictable results is obvious. Therefore the simple substitution of the SMIL playlist format replacing the playlist format disclosed in Belknap would have been obvious, because such a substitution would have produced predictable results in view of the SMIL format's utility for playing videos according to a playlist, as required by the video server of Belknap in view of Zhu.

For **Claim 17**, beyond what was discussed in independent Claim 16, Belknap further teaches:

the video file server of Claim 16, wherein the processor is configured to optionally push (note in Fig. 22 that an operator has the option to selectively distribute media to specific viewers) application specific video content to the at least one media server for display (Col. 10 Lines 39-43, note Element 14 "natively understands the encoding format of the asset" implying appropriate conversion of application specific video to an appropriate display format).

For **Claim 18**, beyond what was discussed in Claim 16, Belknap in view of Zhu teaches:

the video file server of Claim 16, wherein the processor executes the at least one playlist based on logical actions (Belknap: Fig. 32 with Col. 41 Lines 4-9, and Col. 44 Lines 33-57).

The preceding combination of Belknap in view of Zhu does not teach:

the video file server of Claim 16, wherein the processor executes the at least one playlist based on logical actions and *wherein the logical actions depend in part on inputs external to the video file server*.

However, Zhu also teaches:

a remote client can access and manipulate a media object being presented by an application on a remote server (Col. 2 Lines 26-51)

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include inputs external to the media server as taught by Zhu, as logical actions controlling the processor in the execution of the playlist within the video file server, in order to facilitate the ability of a viewer to exercise control over the execution of the playlist from an external location.

For **Claim 19**, beyond what was discussed in Claim 18, Belknap in view of Zhu teaches:

the video file server of Claim 18, wherein the inputs external to the video file server are mapped into application specific commands according to the format of the video file (Zhu: Fig. 2 Elem. 212 with Col. 4 Lines 33-39, and Col. 5 Lines 1-13).

For **Claim 23** Belknap teaches:

a method of distributing video information, comprising:

from a first network location (Fig. 1 Element 12), configuring a playlist of video files (Fig. 32, also Col. 5 Lines 20-26 and Col. 44 Lines 38-57), the video files being stored in at least one second network location (Fig. 1 Elements 18 and 22; and Col. 8 Lines 30-35);

from the second network location, executing a playlist (Col. 44 Lines 49-57), wherein executing includes:

pushing the video content (Col. 8 Lines 54-58) to a third network location (Fig. 1 Elem. 14) according to the playlist (Col. 44 Lines 38-57); and

from the third network location, translating the video content into a video output signal suitable for display (Col. 8 Lines 66-67 through Col. 9 Lines 1-2, and Col. 10 Lines 30-35).

Belknap does not expressly teach:

executing a playlist from a second network location includes:

accepting application specific video content associated with a video file identified in the playlist,

translating the video content to application independent video content and pushing the video content to a third network location according to the playlist.

Zhu teaches:

a document server converting a document from its native form to another form suitable for serving the document (Fig. 2 Elements 102 and 106, with Col. 4 Lines 53-67).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to transform media from a native format (which may not be compatible to the server's clients) into a format suitable for distribution, as taught by Zhu, in the media system taught by Belknap, in order to prevent a client device from being unable to display a non-supported media type.

Belknap in view of Zhu does not expressly teach that the track includes at least one logical action related to playing the playlist.

The SMIL specification describes a playlist format (see the playlist in section 11.1.1) wherein playlist tracks may include logical actions related to playing the playlist (11.1.1 playlist: a duration attribute comprises play and stop actions. See also accelerate, decelerate, autoReverse, and speed attributes in section 11.1.2).

The simple substitution of one known element for another to achiever predictable results is obvious. Therefore the simple substitution of the SMIL playlist format replacing the playlist format disclosed in Belknap would have been obvious, because such a substitution would have produced predictable results in view of the SMIL format's utility for playing videos according to a playlist, as required by the video server of Belknap in view of Zhu.

For **Claim 24**, beyond what was discussed in independent Claim 23, Belknap further teaches:

executing the playlist further includes executing logical actions associated with initiation of display and termination of display of the video files (Fig. 32 Element 1242 and Col. 44 Lines 38-57; also Fig. 24).

For **Claim 25**, beyond what was discussed in Claim 24, Belknap in view of Zhu teaches:

the method of Claim 24, wherein executing logic actions includes the second location receiving external inputs that are mapped into application specific commands (Zhu: Fig. 2 Element 212 with Col. 4 Lines 33-39, and Col. 5 Lines 1-13, note Zhu teaches that a server maps commands to a specific application serving the media).

For **Claim 26**, beyond what was discussed in Claim 25, Belknap in view of Duso further teaches:

the method of Claim 25, wherein executing logic actions includes the second location receiving logic actions from the first location (Belknap: Fig. 32 Element 1200 and Col. 44 Lines 46-54).

For **Claim 28**, beyond what was discussed in independent Claim 23, Belknap further teaches:

the first network location includes a web client (Fig. 1 Elements 12 and 16, also Col. 5 Lines 10-34).

For **Claim 29**, beyond what was discussed in independent Claim 23, Belknap further teaches:

the second network location includes a video file server (Fig. 1 Elem. 18, also Col. 8 Lines 26-34).

For **Claim 30**, beyond what was discussed in independent Claim 23, Belknap further teaches:

the third location includes a media server (Fig. 1 Elem. 14, and Col. 8 Lines 66-67 through Col. 9 Lines 1-2, and Col. 10 Lines 30-43).

For **Claim 31**, beyond what was discussed in Claim 30, Belknap further teaches:

the first network location includes a computer (Fig. 1 Elem. 12, also Col. 10 Lines 43-67) and configuring a playlist (Fig. 32 Element 1200) includes:

downloading an existing playlist from the video file server at the second network location to the computer (Col. 5 Lines 35-67);

editing the playlist (Col. 6 Lines 4-10); and

uploading the edited playlist from the computer to the video file server (Col. 44 Lines 38-57).

Claims 7, 20-22 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Belknap (US Patent 6,763,377), in view of Zhu (US Patent 6,763,501), the SMIL playlist format (SMIL 2.0 W3C Recommendation), and Duso (U.S. Patent 5,892,915).

For **Claim 7**, beyond what was discussed in Claim 6, Belknap in view of Zhu teaches:

logical actions controlling the execution of the playlist are configured using the web client (Belknap: Col. 44 Lines 33-57).

Belknap in view of Zhu does not expressly teach:

the system of Claim 6, wherein the logical actions are configured at least in part in *real time* by a user using the web client.

Duso teaches:

an interface that allows a playlist to be edited after it has been uploaded to a video server (Col. 2 Lines 47-60).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the capability to configure playlists in real time as taught by Duso, within the video system, in order to make adjustments to a playlist that is executing.

For **Claim 20**, beyond what was discussed in Claim 19, Belknap in view of Zhu teaches:

a video file server (Belknap: Fig. 1 Element 18) with the ability to receive application specific commands (Zhu: Col. 5 Lines 1-4).

Belknap in view of Zhu does not expressly teach:

the video file server of Claim 19, wherein the application specific commands include any combination from the set of Play, Restart, Pause, Stop, Rewind, Fast Forward, Next File, Next Slide, Previous Slide, Mouse Click, Hyperlink and Go To New Playlist.

Duso teaches:

application specific commands include:

Play, Restart and Pause (Duso: Col. 37 Lines 45-50, 57-60, and Col. 38 Lines 10-20).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include the application specific commands

taught by Duso, within the video file server, in order to allow a viewer receiving pushed content to exercise control over the playback of said content.

For **Claim 21**, beyond what was discussed in Claim 20, Belknap in view of Zhu, further in view of Duso teaches:

inputs external to the video file server include messages received from the network (Duso: Col. 37 Lines 18-25).

For **Claim 22**, beyond what was discussed in Claim 20, Belknap in view of Zhu, further in view of Duso teaches:

the inputs external to the video file server (Duso: Col. 37 Lines 45-50, 57-60, and Col. 38 Lines 10-20) include a prompt (Belknap: Col. 2 Lines 40-44).

For **Claim 27**, beyond what was discussed in Claim 25, Belknap in view of Zhu teaches:

executing logic actions includes the second location receiving external inputs that are mapped into application specific commands (Zhu: Fig. 2 Element 212 with Col. 4 Lines 33-39, and Col. 5 Lines 1-13, note Zhu teaches that a server maps commands to a specific application serving the media).

Belknap in view of Zhu does not expressly teach:

the method of Claim 25, wherein the application specific commands include any combination from the set of Play, Restart, Pause, Stop, Rewind, Fast Forward, Next File, Next Slide, Previous Slide, Mouse Click, Hyperlink and Go To New Playlist.

Duso teaches:

application specific commands include:

Play, Restart and Pause (Col. 37 Lines 45-50, 57-60, and Col. 38 Lines 10-20).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include the application specific commands taught by Duso, within the method of distributing video information, in order to allow a viewer receiving pushed content to exercise control over said content.

(10) Response to Argument

Appellant argues against the rejections using the combination of Belknap, Zhu, and the SMIL specification as presented above.

Appellant first addresses the combination with the SMIL specification, arguing that not all of the elements of claims 1, 16, and 23 are met. Br. at 13–14. Appellant cites to claim language concerning playlists and then argues that the SMIL specification does not teach that a playlist track includes at least one logical action. Br. at 13. Appellant argues that the SMIL specification provides for interactive controls and simulated motions of an image rather than logical actions. Br. at 14.

Turning first to Appellant's specification, a playlist includes a track comprising an identifier to select a video file and may optionally further include logical actions relating to playing the video file. Para. 0022 in conjunction with Fig. 2. Logical actions include a duration for playing the video file. Id. Other logical actions include start, stop, rewind, fast-forward, etc. Id.

Turning to Belknap, an administrator terminal schedules delivery of a list of assets 1208 comprising video tracks to be delivered to a device at destination 1240. Fig. 32. The list of video tracks comprises video file identifiers, Fig. 32 (e.g. "Pulp.mpg"), and is thus a playlist by Appellant's description. The terminal can further configure logical actions such as duration 1242, repeat 864, etc. However, these logical actions are not part of the playlist as claimed.

Turning to the SMIL specification, a sequence of contents includes several video tracks comprising video files. Ch. 11.1.1, pg. 3 (see the list between the <seq> tags, specifying video files, e.g. "movie1.mpg"). Each video track has a duration element ("dur=10s"), which is a logical action as defined above. The tracks play in sequence according to the list. Id. Other logical actions are cited in the rejections, including a speed attribute, which corresponds to the rewind and fast-forward logical actions described in Appellant's specification. Therefore, the SMIL specification describes a playlist including logical actions as claimed.

Appellant's argument that the combination does not provide the playlist format for a remote device (Br. at 14) is thus unpersuasive since Belknap teaches supplying a playlist to a remote device at destination 1240, Figure 32, as described above. Combining, as described in the rejections, Belknap's delivery of a playlist to a remote device with the SMIL specification's playlist including logical actions, one arrives at the limitations discussed above. Therefore, the combination was properly applied to reject claims 1, 16, and 23.

Appellant next addresses the same combination as applied to dependent claim 31. Br. at 14. Appellant agrees that Belknap downloads a web document comprising scheduling information for editing, but argues that this does not include a playlist. Id.

This is unpersuasive because Belknap's scheduling information is a playlist as previously discussed. A user of the administrator terminal can edit any aspect of the global web document, including the scheduled list of assets. Fig. 32. Since the global web document is downloaded from the video file server 18 to

the administrator terminal 12 (see Fig. 1 in conjunction with col. 48, lls. 11–18) and edited there, Fig. 32, the combination as applied to claim 31 meets the claim limitations.

Appellant argues the provide rationale for combining the SMIL specification with Belknap. Br. at 15. Appellant argues that the SMIL specification relates to "direct presentations" which are not useful for Belknap. Id.

Appellant gives no further indication of what is meant by "direct presentations." The examiner sees no reason to confine the SMIL specification's playlist format to purely local use (if this is what is meant by "direct") when Belknap clearly teaches that playlists can be transmitted remotely across networks, and the SMIL specification gives no indication otherwise.

Appellant further argues that Belknap does not explicitly state that the media server can perform the interactive operations required by the SMIL specification, so one of ordinary skill would not be led to the combination. Br. at 15.

Belknap already teaches performing logical actions on the scheduled media assets. Fig. 32 (see the duration element 1242 including repeat element 864, interval element 886, etc.). These logical actions could have been added into the playlist as taught by the SMIL specification. The examiner finds no requirement in the SMIL specification that "interactive actions" other than the above logical actions must be supported by a media server implementing the specification. Nor can Belknap's lack of an explicit description of support for these "interactive actions" reasonably be said to teach away from the above

combination concerning logical actions. Therefore, the arguments against the combination are unpersuasive.

Appellant argues the combination of Zhu and Belknap. Br. at 16. Appellant argues that Zhu's commands are not transmitted to a server as in Belknap.

Belknap sends commands from an administrator terminal to a video file server to deliver video to an end user terminal. Abstract. Thus the device sending the commands and the device receiving the video responsive to the commands are different. In Zhu, the client device both sends commands and receives the video responsive to the commands. Fig. 9 (the client device sends commands 904 via an application 210 to server applications 216, 212 via intermediate server 102. Screen data 906 responsive to the commands is returned to the client. See further Fig. 2 and description). The examiner thus agrees that there is a difference in the number of devices between the references, Br. at 16.

However, the references both return video data to the client by streaming from a remote server. Belknap Abstract; Zhu Fig. 9 (screen data is for a video display, see col. 5, lls. 28–42). A virtual display driver is used so that the client does not have to understand the application specific data, but can merely render the screen on a video display. Zhu col. 2, l. 60 – col. 3, l. 10. These advantages do not depend on which device the commands to display the data came from, and so the distinction pointed out by Appellant does not prevent Zhu's teaching from being applied to Belknap. Zhu's teaching of an application-independent video driver for streaming video data is thus applicable to Belknap's teaching of a video server for streaming video data, and would have improved Belknap by

allowing clients to display the streaming video without needing the specific video decoder or application to decode/interpret the video files. Therefore, Appellant's arguments against the combination with the Zhu reference are unpersuasive.

Appellant next argues against the rejections using the combination presented above further in view of the Duso reference. Appellant argues that Duso does not make up for the alleged deficiencies of the above references with respect to claims 1, 16, and 23. Br. at 16, 17. However, since the rejections of these claims are not deficient and Duso was not applied in the rejections, this argument is not relevant.

Appellant argues the combination with Duso as applied to claim 7. Br. at 17. Appellant again points out that in Belknap the web client transmitting commands and the media server receiving video are separate devices and contrasts this with Duso, where the transmitting and receiving devices are the same. Id. The benefit of Duso's method for remotely modifying a playlist as it is playing, col. 2, lls. 47–60, does not depend on which device is doing the modifying, but rather lies in the simple fact that real time playlist editing becomes possible. Thus Belknap's administrator could insert new video clips into the playlist at will, Duso col. 3, lls. 5–13, rather than being constrained to scheduling items in advance. The difference in the number of devices between the references does not prevent Duso's teaching from being applied to Belknap. Appellant's argument here is unpersuasive.

Appellant finally argues that Duso has a doubly-linked playlist that is incompatible with the SMIL format. Br. at 18. However, the doubly-linked list

implementation is merely exemplary, Duso col. 33, lls. 14, 15. The playlists only require a few simple elements, namely a listing of video tracks and associated logical actions, for which many data structures could be used. The examiner finds no teaching that the playlists of the references could not be combined by one of ordinary skill in the art. This argument does not overcome the prima facie case of obviousness already presented.

Therefore, Applicant's arguments are unpersuasive in total, and the rejections should be sustained.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Bennett Ingvoldstad/

Examiner, Art Unit 2427

Conferees:

/Bennett Ingvoldstad/

Examiner, Art Unit 2427

/Scott Beliveau/

Supervisory Patent Examiner, Art Unit 2427

/Jason P Salce/

Primary Examiner, Art Unit 2421